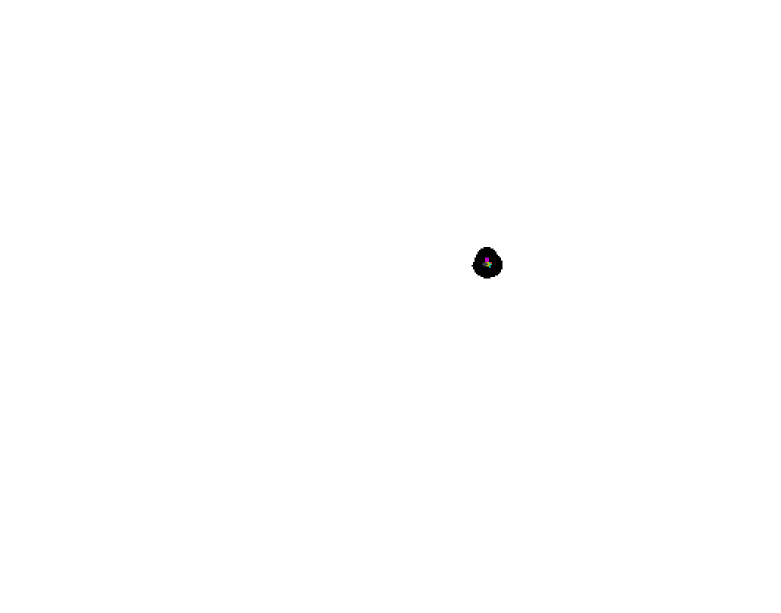
RANDOM WALKS - DIFFUSION MODEL

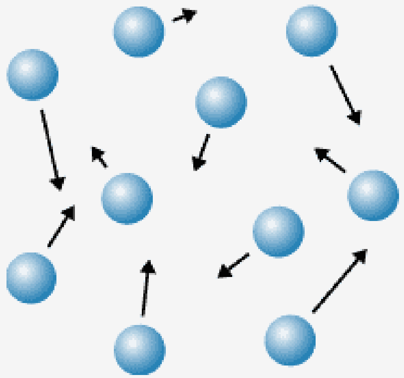
Natalia Wołowiec
Kacper Kalinowski
Mateusz Miller

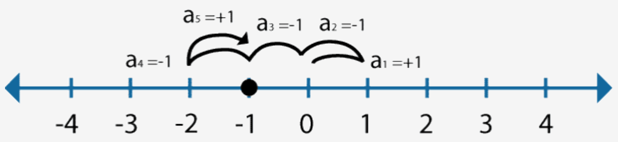
Table of contents

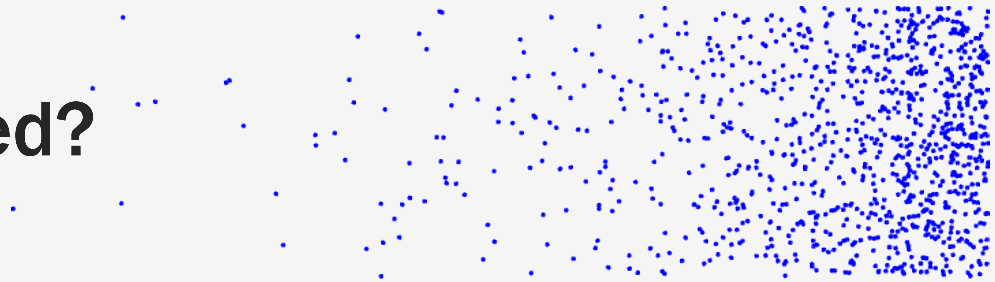


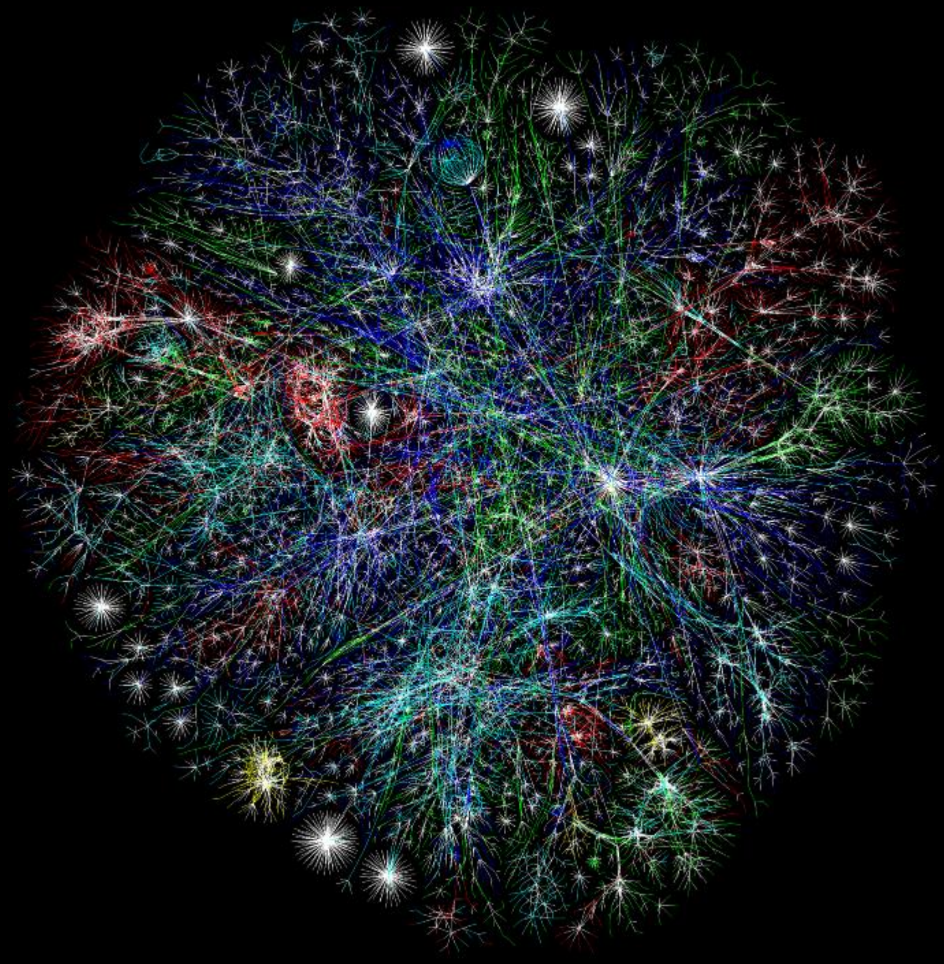
Random walk

1. What is it?
2. In real life?
3. Math behind it
4. Variable step size?
5. Biased?
6. 1D – 2D – 3D

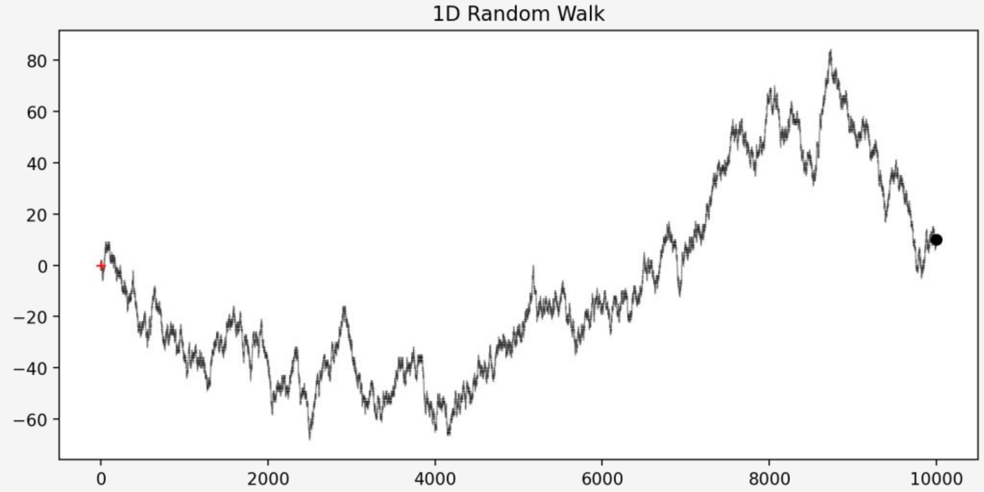


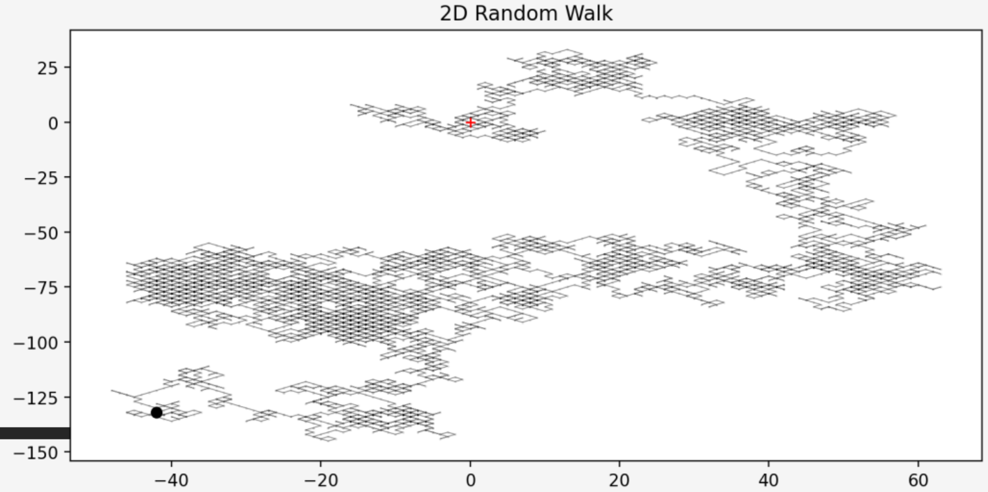


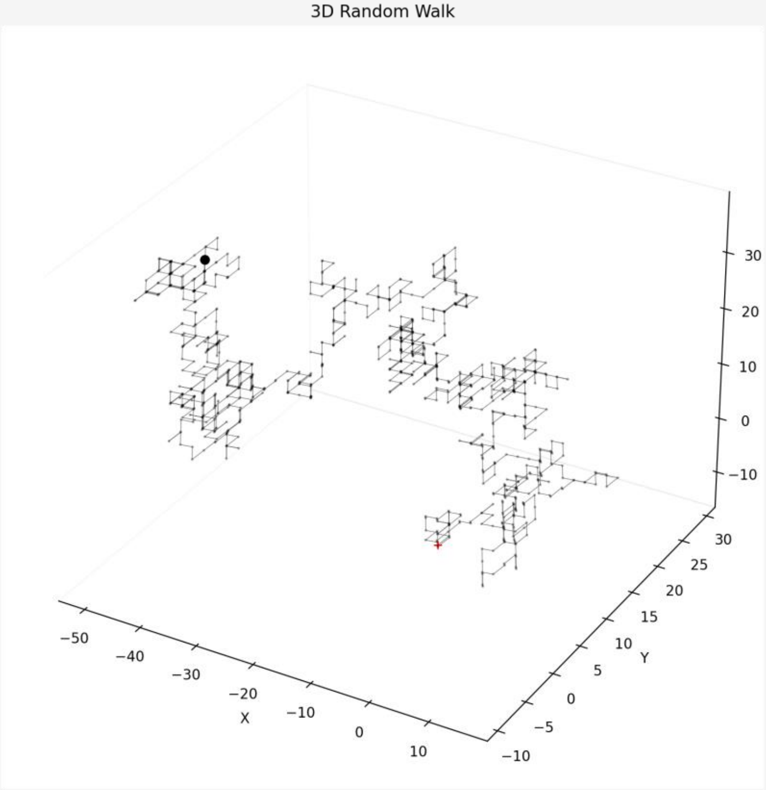




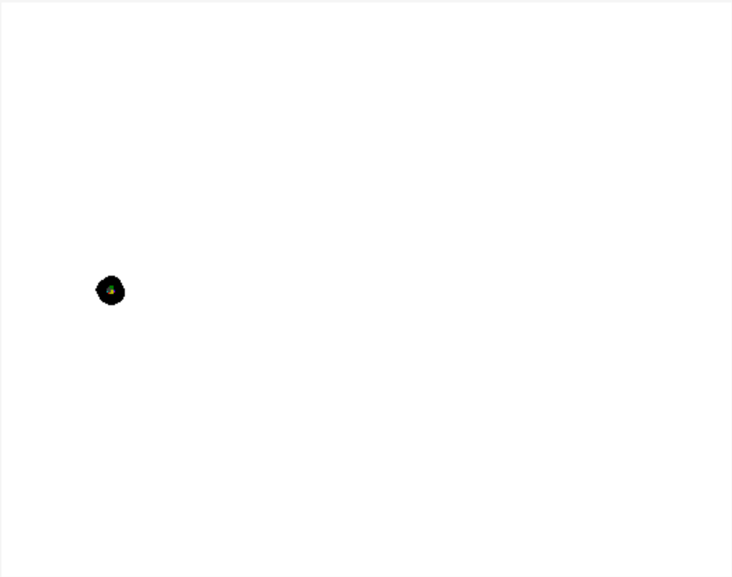
Graphs and pictures



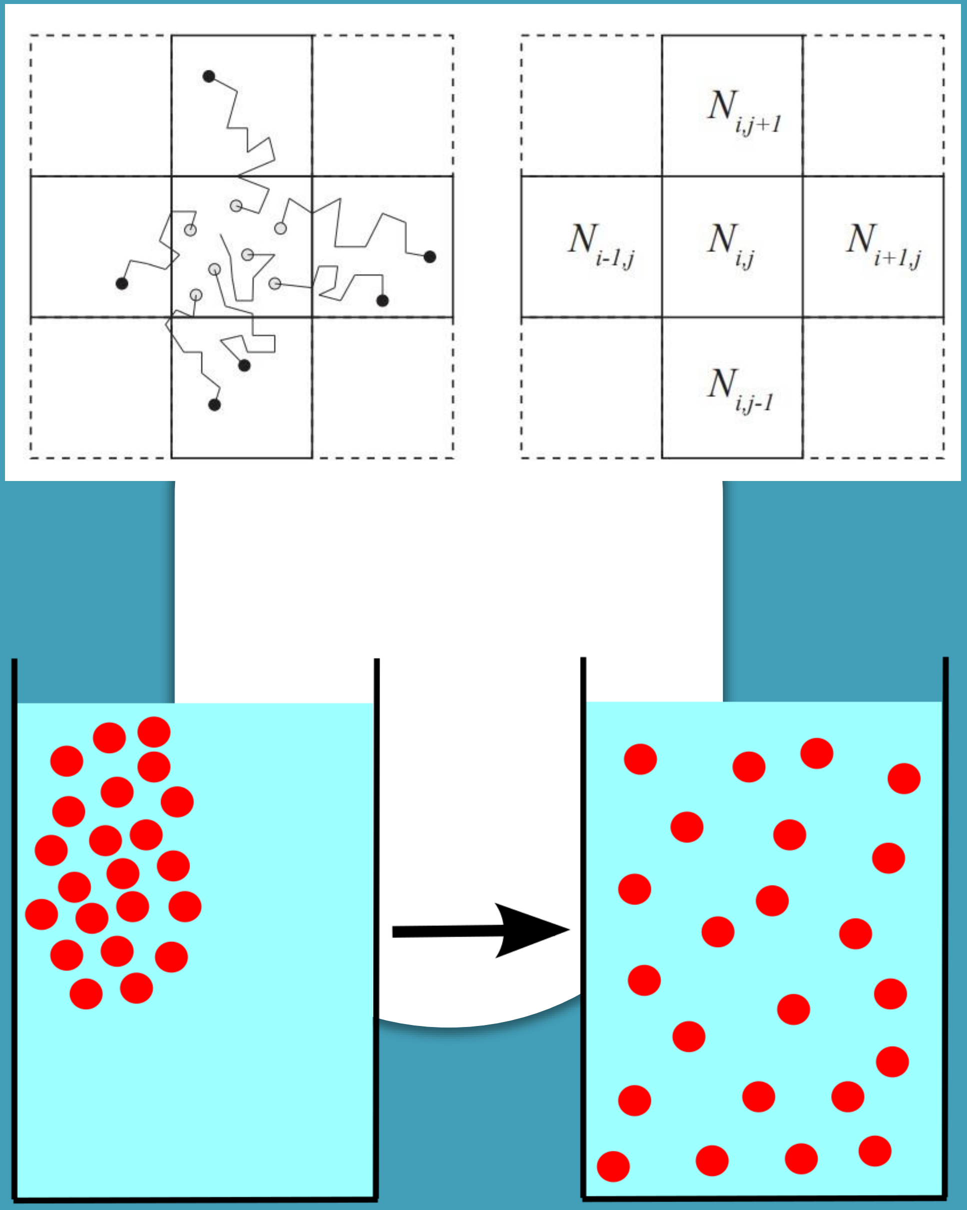




Biased walk toward right
(with step size)



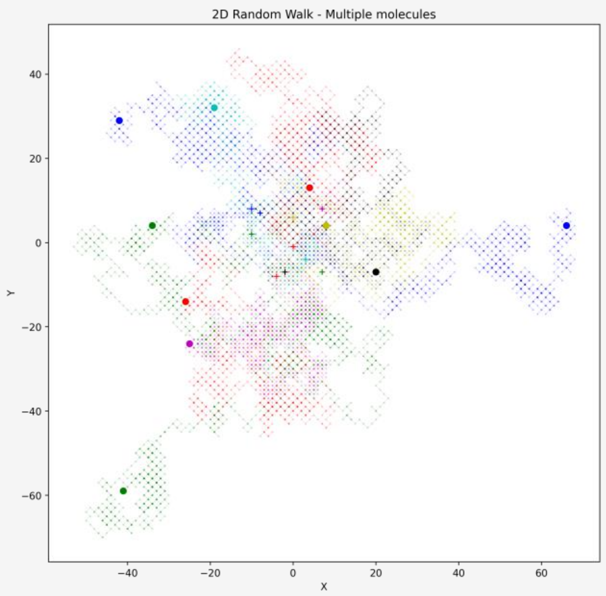
Diffusion

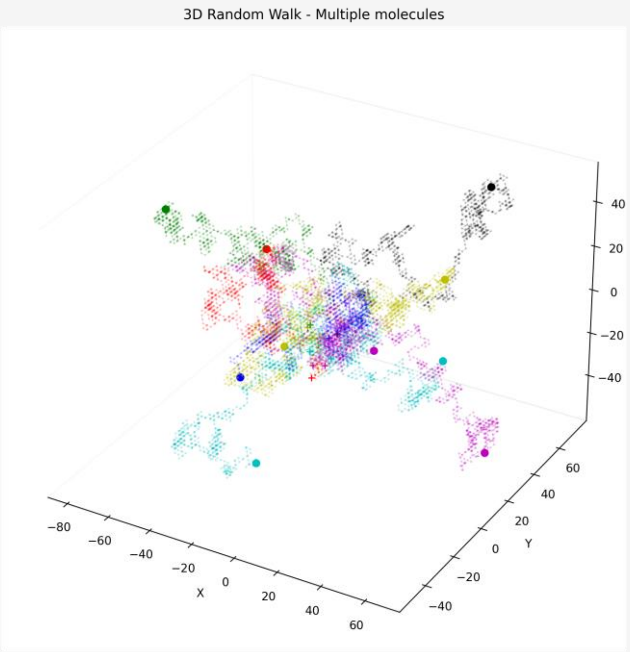


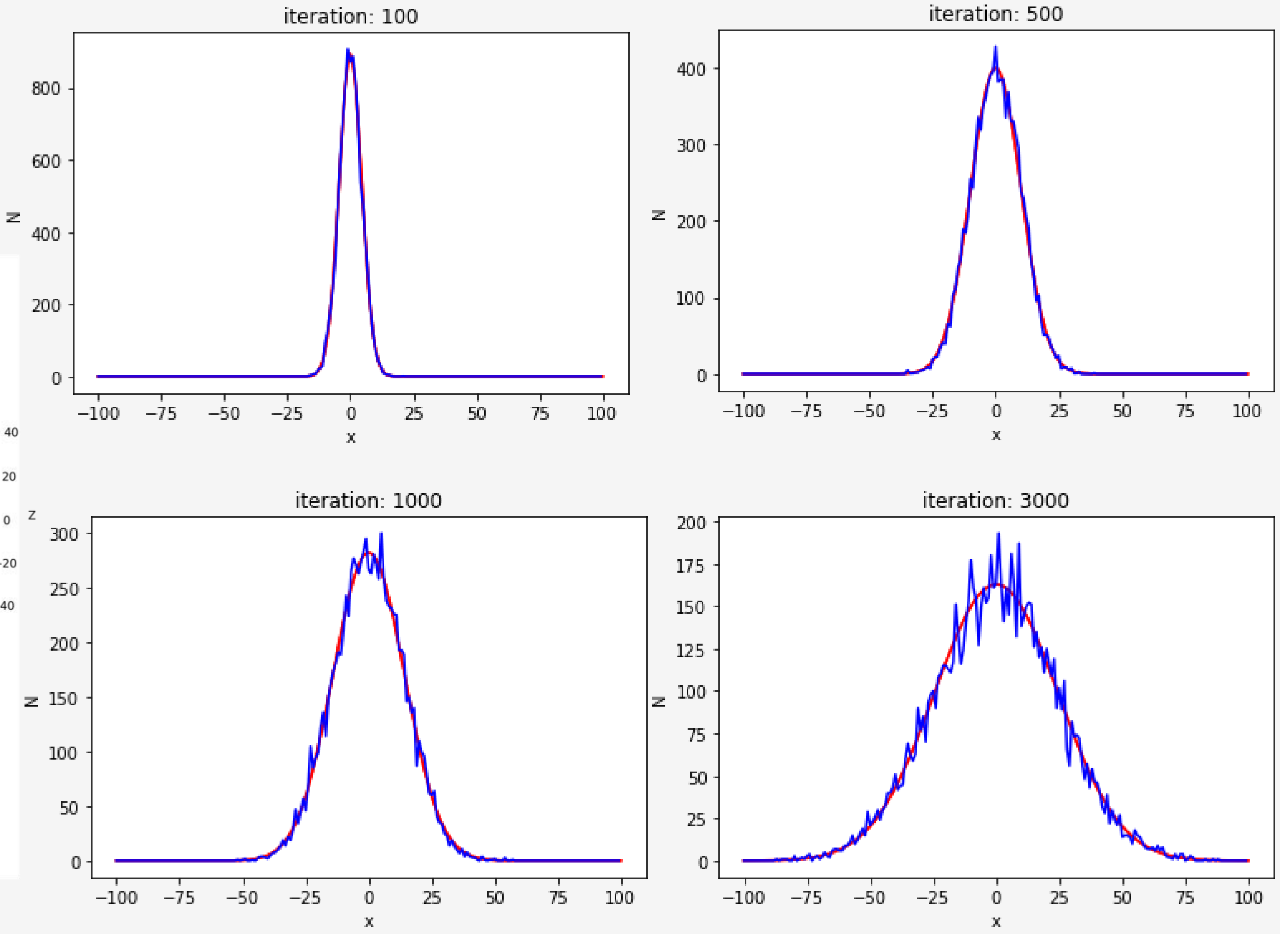
1. Analysis in one dimension
2. Two aprroaches
3. Comparing
4. Equation
5. Efficient?

$$\frac{\partial c}{\partial t} = D\left(\frac{\partial c^2}{\partial x^2} + \frac{\partial c^2}{\partial y^2}\right) = D\nabla^2 c$$

Graphs and pictures v2







Python

Comparing distribution of walkers for the diffusion model and random walkers in 1D

```
M = 10000 # Nr of walkers
L = 100 # Max size of lattice
# Each time step - move walkers and propagate diffusion solution
p = 0.1 # Prob for motion
pinv = 1.0-p
nsteps = 3001 # Nr of timesteps
# Initialize walkers
x = np.zeros(M) # Initial position of walkers
edges = np.array(range(-L,L+1))-0.5
xc = 0.5*(edges[:-1]+edges[1:])
# Initialize concentrations
c = np.zeros((2*L+1,2))
i0 = 0
i1 = 1
c[L] = M # c[L] corresponds to x = 0
cx = range(-L,L+1)
D = p
plt.ion()
noutput = 100
for it in range(nsteps):
    # First update positions of all random walkers
    for iw in range(M):
        rnd = np.random.rand(1)
        dx = -1*(rnd<p)+1*(rnd>pinv)
        x[iw] = x[iw] + dx
        # Perform explicit step for diffusion equation
    for ix in range(1,len(c)-1):
        # use i0 and generate i1
        c[ix,i1] = c[ix,i0] + D*(c[ix-1,i0]-2*c[ix,i0]+c[ix+1,i0])
    # Flip i0 and i1
    ii = i1
    i1 = i0
    i0 = ii
    # Plot the two concentrations
    if it in [100,500,1000,3000]:
        Nx,e = np.histogram(x,edges)
        plt.clf()
        plt.plot(cx,c,'-r',xc,Nx,'-b')
        plt.title(f'iteration: {it}')
        plt.xlabel('x'),plt.ylabel('N'),plt.pause(0.001)
```

Our code

- https://github.com/Sheaza/random-walk-pfe-project/blob/main/random-walk.ipynb?fbclid=IwAR3KwjN0D65NQ01cxubP-cgSuRzFS46V9Pf71b8ypnSSkNfjMEgVYHFqD0U

Bibliography

- https://www.princeton.edu/~akosmrlj/MAE545_S2018/lecture17_slides.pdf?fbclid=IwAR26zLYwCp1iAUv-qBStrOt_PPKu-IB3jMFyInjCwrmT0tluuq0mIEu-dW0
- https://www.mit.edu/~kardar/teaching/projects/chemotaxis(AndreaSchmidt)/random.htm?fbclid=IwAR14J_OFXDJkNBvkr_u9fIvJlqOnIxwy8fUKTbuDPXMoy5mx11ertIAeno0
- https://towardsdatascience.com/random-walks-with-python-8420981bc4bc
- https://www.uio.no/studier/emner/matnat/fys/FYS2160/h17/simuleringsopgaver/virrevandrer_diffusjon.pdf?fbclid=IwAR19ROOil_U2m9p-g-ZDUe_fzBEa9NotBIVPVtu3_WIWpw-E3CZU3uWbODA

THANK YOU FOR YOUR ATTENTION!

Natalia Wołowiec
Kacper Kalinowski
Mateusz Miller